A brief survey of Adversarial Machine Learning and Defense Strategies

Zahid Akhtar[1] and **Dipankar Dasgupta**[2]

[1]Research Assistant Professor, Center for Information Assurance
Department of Computer Science
The University of Memphis
Memphis, Tennessee, USA
Email: zmomin@memphis.edu

[2]Director, Center for Information Assurance
Hill Professor of Cybersecurity
The University of Memphis
Memphis, Tennessee, USA
Email: dasgupta@memphis.edu

Technical Report No. CS-19-002 December 1, 2019

Abstract:

This brief survey compiled different adversarial attacks that are reported, and associated defense strategies devised. While all Machine Learning (ML) techniques are not Neural Networks (NN) or Deep Learning (DL) but many scholars and practitioners are using the terms interchangeably. This summary uses the term ML for generalization. The majority of Adversarial Machine Learning (AML) focused on image manipulations (and crafted attacks are specific to datasets), some works studied other media and tried to develop universal perturbations schemes. Also, reported works in the literature on defense mechanisms provide piece-meal solutions to AML. Whereas, this survey provides the information in a concise tabular form (highlighting important features, strategies, and classification) for better understanding and clarity.

.

Overview:

Machine Learning (ML) techniques have recently attained impressive performances on diverse and challenging problems such as malware/intrusion detection, image classification, object detection, speech recognition, face recognition in-the-wild, self-driving vehicles, just to name a few. In spite of their major breakthroughs in solving complex tasks, it has been lately discovered that ML techniques (especially artificial neural networks and data-driven artificial intelligence) are highly vulnerable to deliberately crafted samples either at training or at test time, which can easily subvert ML techniques' outcomes. The samples with deliberate perturbations are usually referred as '*adversarial examples*' (a.k.a. wild pattern or adversarial attack), i.e., carefully-perturbed samples aimed to mislead the ML techniques. For instance, the arbitrary perturbations added in the benign malware binary vector/file can lead to a significant drop in accuracy of DNNs-based malware detection systems. Similarly, for image classification ML techniques, an adversarial example can be generated by adding some indiscernible perturbations into a given image. The resultant adversarial image is misclassified by the well-known ML classifiers, while a human being can still classify it correctly without spotting the deliberate added perturbations. In case of automatic speech-to-text transcription, a small perturbation (e.g., an arbitrary waveform) when added to the original waveform can cause it to be transcribed as any phrase malicious adversary chooses. The audio adversarial examples that are perceived one way by a human but transcribed differently by a state-of-the-art speech-to-text transcription neural network. Also, an adversary can malignly modify labels of the samples to be used for (re-)training of ML techniques, which is known as poisoning attacks.

To safeguard ML techniques against malicious adversary, several countermeasure schemes have been proposed, which roughly fall within two categories: adversarial defense and adversarial detection. Frameworks in first category aim at improving the DNNs' robustness to classify AEs correctly, e.g., adversarial training, i.e., training the ML techniques with clean and malicious samples. While the frameworks in second category attempt to detect malicious samples before they are fed to ML technique's main architecture such as augmenting the ML technique's main model with a small "detector" sub-ML technique trained on both adversarial and original clean samples, which can be utilized to distinguish whether the input sample is an adversarial attack or not. Despite the current progress on increasing robustness of ML techniques against malicious attacks, majority of existing countermeasures still do not scale well and have low generalization. Namely, adversaries (adversarial samples/input) yet pose great threats to machine learning (ML) and artificial intelligence (AI) [40].

Adversarial Input/Sample Generation Methods:

There exists a several methods to generate adversarial samples or adversarial examples (AEs). We briefly describe the standard and representative adversarial attack generation methods utilized in image, text and audio domains. Adversarial attacks to deep learning-based systems can be either black-box or white-box, where the adversary, respectively, does not have and has knowledge of the model architecture, parameters and its training data. In addition, the attacks could be targeted and non-targeted, that aim to misguide DNNs to a specific class and arbitrary class except the correct one, respectively. Following, we outline main adversarial example generation techniques, which are also summarized in Table 1.

Fast Gradient Sign Method (FGSM) [1]: Elements of attack are generated by taking one step update along the sign of gradient of a loss function essential to the sample.

Iterative Gradient Sign Method (IGSM) [2]: It is iterative version of FGSM, which takes multiple small steps iteratively while adjusting the direction after each step.

Jacobian Saliency Map Attack (JSMA) [3]: Attacks are generated by discovering the importance of each pixel in the decision process such as a saliency map is generated by computing the forward derivative (Jacobian) of the function learned by a DNN.

DeepFool (DF) [4]: It finds the closest distance from the original input to the decision boundary of AA. To overcome the nonlinearity in high dimension, it performs an iterative attack with a linear approximation.

One-Step Target Class Method (OSTCM) [5]: It is extension of FGSM to a targeted attack by maximizing the probability of the target class.

Basic Iterative Method (BIM) [2]: It is AA to the physical world. It is extension of the FGSM by running a finer optimization (smaller change) for multiple iterations.

Iterative Least-Likely Class Method (ILLC) [2]: It is BIM attack to a specific class by choosing the least-likely class of prediction and tried to maximize the cross-entropy loss.

Compositional Pattern-Producing Network-Encoded Evolutionary Algorithm (CPPN EA) [6]: It utilizes evolutionary algorithm (EA) to yield the AA. To solve a multiclass classification problem using EA, the method applies multidimensional archive of phenotypic elites MAP-Elites.

Carlini and Wagner's Attack (C&W) [7]: It employs L_2, L_0, and L_∞ algorithms that generate AAs causing misclassification with the same label.

Universal Perturbation (UP) [9]: It is a universal image-agnostic perturbation attack method that fools classifiers by single adversarial perturbation to all images.

Feature Adversary (FA) [11]: This method minimizes the distance of the representation of internal neural network layers instead of the output layer to produce AA.

Hot/Cold method (H/C) [12]: This method finds multiple AA for every single image input. It first aligns the modified image with the original image (cold) and then measure the similarity between the perturbed image (hot).

Model-based Ensembling Attack (MEA) [14]: This technique can generate transferable AAs to attack many DNNs using ensemble-based approaches.

Ground-Truth Attack (GTA) [15]: It conducts a binary search and finds AAs with the smallest perturbation by invoking Reluplex iteratively.

Targeted Audio Adversarial Examples (TAAE) [25]: It is an iterative optimization-based targeted attack to a state-of-the-art speech-to-text transcription neural network via optimization based on the MFC pre-processing transformation.

Zeroth Order Optimization (ZOO) [8]: It does not require gradients and utilizes hinge like loss function and symmetric difference quotient to generate AA.

One Pixel Attack (OPA) [10]: To avoid the problem of measurement of perceptiveness, this technique generates AAs by only modifying one pixel based on differential evolution.

Natural GAN (NGAN) [13]: It utilizes generative adversarial networks (GANs) that minimizes the distance of the inner representations to generate AAs.

Zero-Query Attacks (ZQA) [16]: It trains a surrogate model on the same task as the target model, performs a gradient-based attack on the surrogate model, and replays this generated AA on the target model.

Natural Evolution Strategies (NES) [17]: It uses gradient estimation technique and employs projected gradient descent with the estimated gradient to construct AAs.

Boundary Attack (BA) [18]:It is a gradient-free AA that starts with an image of the target class and then makes steps alternating between moving the image along the decision boundary (while remaining adversarial) and steps which move towards the original image.

Greedy Search Algorithm (GSA) [19]: It is an iterative procedure that considers at each step all valid one-word changes to improve the AAs by applying greedy optimization strategy.

Genetic Attack (GA) [20]: It exploits population-based gradient free optimization via genetic algorithms to replace words with their synonyms so as to generate semantically and syntactically similar AAs.

Improved Genetic Algorithm (IGA) [21]: This procedure adopts the genetic metaheuristic for synonyms substitution to attain AAs.

Probability Weighted Word Saliency (PWWS) [22]: It considers the word saliency as well as the classification probability to obtain AAs.

Replacement, Insertion and Removal of Words (RI&RoW) [23]: It is iterative method that combines three different kinds of modifications to alter a regular input into an AA by replacement, insertion and removal of words into the text.

Real-World Noise (RWN) [24]: This technique adds real-world scenario noises such as café, meeting, and station to generate AAs.

Genetic Algorithms and Gradient Estimation (GA&GE) [26]: It combines genetic algorithms and gradient estimation to construct AAs. The attack is first carried out by gradient-free genetic algorithms, then gradient estimation is utilized to determine careful noise placement.

Countermeasures for Adversarial Examples:

The defenses for mitigating AEs can be roughly grouped into seven categories. For instance, adversarial training [27], which is training the system with AEs to augment the regularization and loss functions and making the system more resilient. The other technique is defensive distillation [29] in which additional DNNs with softmax are trained to obstruct the deep learning system from fitting too tightly to the data. Other approaches are pre-processing or denoising [35], i.e., removing the adversarial noise from the input samples before feeding them to neural networks, and model robustifying [21], i.e., modifying the traditional neural network architectures, e.g., adding extra specific robust layers and functions. Also, a category called adversarial examples detection techniques, which focus on detecting AEs as a binary classification or anomaly detection problem. An AEs detector distinguishes whether the input sample is an adversarial attack or not [38]. Table 2 summaries the countermeasures against adversarial examples.

Attack Category	**Adversarial Attack (AA)**	**Acronym Used**
White-Box	Fast Gradient Sign Method (FGSM) [1]♣	AA1
	Iterative Gradient Sign Method (IGSM) [2]♣	AA2
	Jacobian Saliency Map Attack (JSMA) [3]♣	AA3
	DeepFool (DF) [4]♣	AA4
	One-Step Target Class Method (OSTCM) [5]♣	AA5
	Basic Iterative Method (BIM) [2]♣	AA6
	Iterative Least-Likely Class Method (ILLC) [2]♣	AA7
	Compositional Pattern-Producing Network-Encoded Evolutionary Algorithm (CPPN EA) [6]♣	AA8
	Carlini and Wagner's Attack (C&W) [7]♣	AA9
	Universal Perturbation (UP) [9]♣	AA11
	Feature Adversary (FA) [11]♣	AA13
	Hot/Cold method (H/C) [12]♣	AA14
	Model-based Ensembling Attack (MEA) [14]♣	AA16
	Ground-Truth Attack (GTA) [15]♣	AA17
	Targeted Audio Adversarial Examples (TAAE) [25]♦	AA27
Black-box	Zeroth Order Optimization (ZOO) [8]♣	AA10
	One Pixel Attack (OPA) [10]♣	AA12
	Natural GAN (NGAN) [13]♣	AA15
	Zero-Query Attacks (ZQA) [16]♣	AA18
	Natural Evolution Strategies (NES) [17]♣	AA19
	Boundary Attack (BA) [18]♣	AA20
	Greedy Search Algorithm (GSA) [19]#	AA21
	Genetic Attack (GA) [20]#	AA22
	Improved Genetic Algorithm (IGA) [21]#	AA23
	Probability Weighted Word Saliency (PWWS) [22]#	AA24
	Replacement, Insertion and Removal of Words (RI&RoW) [23]#	AA25
	Real-World Noise (RWN) [24]♦	AA26
	Genetic Algorithms and Gradient Estimation (GA&GE) [26]♦	AA28

Table 1: Adversarial Examples Types. Application Domain (Test Environment): ♣Image Domain, #Text Domain, ♦Audio Domain.

Defense Technique	**Approach/Scheme**	**Attacks Studied**
Adversarial Training	Ensemble Adversarial Training, a training methodology that incorporates perturbed inputs transferred from other pre-trained models [27]	AA1, AA2, AA7
	Extended adversarial and virtual adversarial training as a means of regularizing a text classifier by stabilizing the classification function [28]	AA25
	Training the state-of-the-art speech emotion recognition on the mixture of clean and adversarial examples to help regularization [24]	AA26
Defensive Distillation	The main idea used is training the model twice, initially using the one-hot ground truth labels but ultimately using the initial model's probability as outputs to enhance robustness [29] [30]	AA1, AA2
		AA25
Input Reconstruction (Manifold Analysis)	MagNet method transform adversarial examples to clean data via reconstruction. Specifically, it approximates the manifold of normal examples and moves adversarial examples towards the manifold of normal examples to correctly classifying adversarial examples with small perturbation [31]	AA1, AA2, AA4, AA9
Defense-GAN	A framework leveraging the expressive capability of generative models to defend deep neural networks against adversarial attacks [32]	AA1, AA2, AA3, AA4, AA9
Model Robustifying	Synonyms encoding method that inserts an encoder before the input layer of the model and then trains the model to eliminate adversarial perturbations [21]	AA21, AA22, AA23, AA24
	An architecture using Bayesian classifiers (Gaussian processes with RBF kernels) to build more robust neural networks [33]	AA1, AA9
	The proposed strategy used an ensemble of classifiers with weighted/unweighted average of their prediction to increase robustness against attacks [34]	AA1, AA6
Pre-Processing Defense (Transformations Defense)	Using PCA, low-pass filtering, JPEG compression, soft thresholding techniques as pre-processing technique to improve robustness [35]	AA1, AA2, AA9
	Use of use two randomisation operations: (1) random resizing of input images and (2) random padding with zeros around the input images [36]	AA1, AA4, AA9

Adversarial Examples Detection	First, the features are squeezed either by decreasing each pixel’s color bit depth or smoothing the sample using a spatial filter. Then, a binary classifier that uses as features the predictions of a target model before and after squeezing of the input sample [37]	AA1, AA3, AA6, AA9
	A framework that utilizes ten nonintrusive image quality features to distinguish between legitimate and adversarial attack samples [38]	AA1, AA2, AA3, AA4
	Multiversion Programming based an audio AE detection approach, which utilizes multiple off-the-shelf Automatic Speech Recognition systems to determine whether an audio input is an AE [39]	AA27, AA28

Table 2: Summary of countermeasures against adversarial examples.

References:

1. I.J. Goodfellow, J. Shlens, C. Szegedy, Explaining and harnessing adversarial examples, arXiv preprint arXiv:1412.6572, pp. 1-, 2014.
2. A. Kurakin, I. Goodfellow, S. Bengio, Adversarial examples in the physical world, arXiv preprint arXiv:1607.02533, pp. 1-14, 2017.
3. N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami, The limitations of deep learning in adversarial settings, in Proc. IEEE Eur. Symp. Secur. Privacy (EuroS&P), pp. 372–387, 2016.
4. S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, DeepFool: A simple and accurate method to fool deep neural networks, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 2574–2582, 2016.
5. A. Kurakin, I. Goodfellow, and S. Bengio, Adversarial machine learning at scale, Proc. Int. Conf. Learn. Represent. (ICLR), pp. 1–17, 2017.
6. A. Nguyen, J. Yosinski, and J. Clune, Deep neural networks are easily fooled: High confidence predictions for unrecognizable images, Proc. IEEE Conf. Comput. Vis. Pattern Recognit., pp. 427–436, 2015.
7. N. Carlini and D. Wagner, Towards evaluating the robustness of neural networks, Proc. IEEE Symp. Secur. Privacy (S&P), pp. 39–57, 2017.
8. P.-Y. Chen, H. Zhang, Y. Sharma, J. Yi, and C.-J. Hsieh, ZOO: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models, arXiv:1708.03999, pp. 1-13, 2017.
9. S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, Universal adversarial perturbations, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 86–94, 2017.

10. J. Su, D. V. Vargas, and S. Kouichi. One pixel attack for fooling deep neural networks, arXiv:1710.08864, pp. 1-15, 2017.
11. S. Sabour, Y. Cao, F. Faghri, and D. J. Fleet, Adversarial manipulation of deep representations, Proc. Int. Conf. Learn. Represent. (ICLR), pp. 1-18, 2016.
12. A. Rozsa, E. M. Rudd, and T. E. Boult, Adversarial diversity and hard positive generation, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) Workshops, pp. 25–32, 2016.
13. Z. Zhao, D. Dua, and S. Singh, Generating natural adversarial examples, arXiv:1710.11342, pp. 1-15, 2017.
14. Y. Liu, X. Chen, C. Liu, and D. Song, Delving into transferable adversarial examples and black-box attacks, Proc. Int. Conf. Learn. Represent. (ICLR), pp. 1–14, 2017.
15. N. Carlini, G. Katz, C. Barrett, and D. L. Dill, Provably minimally-distorted adversarial examples, arXiv:1709.10207, pp. 1-8, 2017.
16. N. Papernot, P.D. McDaniel, I.J. Goodfellow, S. Jha, Z. B. Celik, and A. Swami, Practical black-box attacks against deep learning systems using adversarial examples, arXiv:1602.02697, pp. 1-14, 2018.
17. A. Ilyas, L. Engstrom, A. Athalye, and J. Lin, Black-box adversarial attacks with limited queries and information, International Conference Machine Learning, pp. 1-10, 2018.
18. W. Brendel, J. Rauber, and M. Bethge, Decision-based adversarial attacks: Reliable attacks against black-box machine learning models, *International Conference on Learning Representations, pp. 1-12*, 2018.
19. V. Kuleshov, S. Thakoor, T. Lau, and S. Ermon, Adversarial Examples for Natural Language Classification Problems, OpenReview submission OpenReview:r1QZ3zbAZ., pp. 1-13, 2018.
20. M. Alzantot, Y. Sharma, A. Elgohary, B. Ho, M.B. Srivastava, and K. Chang, Generating Natural Language Adversarial Examples, Empirical Methods in Natural Language Processing (EMNLP), pp. 2890–2896, 2018.
21. X. Wang, H. Jin, and K. He, Natural language adversarial attacks and defenses in word level, arXiv preprint arXiv:1909.06723, pp. 1-13, 2019.
22. S. Ren, Y. Deng, K. He, and W. Che, Generating Natural Language Adversarial Examples through Probability Weighted Word Saliency, Association for Computational Linguistics, pp. 1-13, 2019.
23. S. Samanta and S. Mehta, Towards crafting text adversarial samples, arXiv preprint arXiv:1707.02812, pp. 1-11, 2017.
24. S. Latif, R. Rana, J. Qadir, Adversarial machine learning and speech emotion recognition: Utilizing generative adversarial networks for robustness, arXiv preprint arXiv:1811.11402, pp. 1-8, 2018.
25. N. Carlini and D. Wagner, Audio Adversarial Examples: Targeted Attacks on Speech-to-Text, *2018 IEEE Security and Privacy Workshops (SPW)*, pp. 1-7, 2018.
26. R. Taori, A. Kamsetty, B. Chu, N. Vemuri, Targeted adversarial examples for black box audio systems, arXiv preprint arXiv:1805.07820, pp. 1-6. 2018.

27. F. Tramer, A. Kurakin, N. Papernot, D. Boneh, and P. McDaniel, Ensemble adversarial training: Attacks and defenses, Proceedings of the International Conference on Learning Representations, pp. 1-20, 2018.
28. T. Miyato, S.I. Maeda, M. Koyama, S. Ishii, Virtual adversarial training: a regularization method for supervised and semi-supervised learning, IEEE transactions on pattern analysis and machine intelligence, vol. 41, no. 8, pp.1979-1993, 2018.
29. N. Papernot, P. McDaniel, X. Wu, S. Jha, A. Swami, Distillation as a defense to adversarial perturbations against deep neural networks, IEEE Symp. on Security and Privacy, pp. 582-597, 2016.
30. M. Soll, T. Hinz, S. Magg, S. Wermter, Evaluating Defensive Distillation for Defending Text Processing Neural Networks Against Adversarial Examples, International Conference on Artificial Neural Networks, pp. 685-696, 2019.
31. D. Meng, H. Chen, Magnet: a two-pronged defense against adversarial examples, ACM SIGSAC Conference on Computer and Communications Security. ACM, pp 135–147, 2017.
32. P. Samangouei, M. Kabkab, R. Chellappa, Defense-gan: Protecting classifiers against adversarial attacks using generative models, arXiv preprint arXiv:1805.06605, pp. 1-17, 2018.
33. J. Bradshaw, A. G. de G. Matthews, and Z. Ghahramani, Adversarial examples, uncertainty, and transfer testing robustness in Gaussian process hybrid deep networks, pp. 1-33, 2017.
34. T. Strauss, M. Hanselmann, A. Junginger, H. Ulmer, Ensemble Methods as a Defense to Adversarial Perturbations against Deep Neural Networks, arXiv:1709.03423, pp. 1-10, 2018.
35. Y. Yamada, E. Weinberger, A. Cloninger, X. Cheng, K. Stanton, U. Shaham, J. Garritano, Y. Kluger, Defending against Adversarial Images using Basis Functions Transformations. arXiv preprintarXiv:1803.10840, pp. 1-12, 2018.
36. C. Xie, J. Wang, Z. Zhang, Z. Ren, and A. Yuille. Mitigating adversarial effects through randomization. arXiv preprint arXiv:1711.01991, 2017.
37. W. Xu, D. Evans, and Y. Qi, Feature squeezing: Detecting adversarial examples in deep neural networks, arXiv:1704.01155, pp. 1-15, 2017.
38. Z. Akhtar, J. Monteiro and T. Falk, Adversarial Examples Detection Using No-Reference Image Quality Features, IEEE International Carnahan Conference on Security Technolog, pp. 1-5, 2018.
39. Z. Qiang, et al., A multiversion programming inspired approach to detecting audio adversarial examples, 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), pp. 1-10, 2019.
40. Dipankar Dasgupta. AI vs. AI: Viewpoints. Technical Report (No. CS-19-001), The University of Memphis, May 2019.

About the Authors:

Dr. Zahid Akhtar is currently a research assistant professor at the University of Memphis, USA. He is a senior member of IEEE and member of ACM. He received the Ph.D. degree in electronic and computer engineering from the University of Cagliari, Italy. He was a Postdoctoral Fellow with INRS-EMT, University of Quebec, Canada, University of Udine, Italy, Bahcesehir University, Turkey, and the University of Cagliari. His research interests include the areas of computer vision and machine learning with applications to biometrics, affect recognition, image and video processing, audiovisual multimedia quality assessment, and cybersecurity. Dr. Akhtar has published 6 book chapters, 28 journal articles and 48 conference papers (+830 citations as per google scholar, 2019). He has received the Premium Award for Best Paper in IET Biometrics Journal (2014), Outstanding Paper Award at the International Conference on Internet (2017), Best Industry-Oriented research work award at National Seminar on Physics and Technology of Sensors (2007), Outstanding Contribution in Reviewing Award in the prestigious journal Pattern Recognition Letters (2016), and Thrice Best Reviewer Award at the International Conference on Vision, Image and Signal Processing (2017, 2018 & 2019).

Dr. Dipankar Dasgupta is a professor of Computer Science at the University of Memphis since 1997, an IEEE Fellow and an ACM Distinguished Speaker. Dr. Dasgupta is known for his pioneering work on the design and development of intelligent solutions inspired by natural and biological processes. During 1990-2000, he extensively studied different AI/ML techniques and research in the development of an efficient search and optimization method (called structured genetic algorithm) has been applied in engineering design, neural-networks, and control systems. He is one of the founding fathers of the field of artificial immune systems (a.k.a Immunological Computation) and is at the forefront of applying bio-inspired approaches to cyber defense. His notable works in digital immunity (featured in Computer World Magazine 2003), negative authentication, cloud insurance modeling and adaptive multi-factor authentication demonstrated the effective use of various AI algorithms.

Dr. Dasgupta has authored four books, his latest graduate textbook is on Advances in User Authentication published by Springer-Verlag, August 2017. In addition, Dr. Dasgupta has published more than 260 research papers (+16,000 citations as per google scholar) in book chapters, journals, and international conference proceedings. Among many awards, he was honored with the 2014 ACM-SIGEVO Impact Award for his seminal work on negative authentication, an AI-based approach. He also received five best paper awards in different International conferences and has been organizing IEEE Symposium on Computational Intelligence in Cyber Security at SSCI since 2007. Dr. Dasgupta speaks regularly, serves as panelist and keynote speaker and offer tutorials in leading computer science conferences, and have given more than 300 invited talks in different universities and industries.